

EAST Search History

Ref #	Hits	Search Query	DBs	Default Operator	Plurals	Time Stamp
S56	1	"20050060293"	US-PGPUB; USPAT	OR	OFF	2006/08/22 17:07
S57	2	"6757670"	US-PGPUB; USPAT	OR	OFF	2006/08/23 11:57
S58	61	"5560007"	US-PGPUB; USPAT	OR	OFF	2006/08/23 12:21
S59	21	(concurrent\$4 or while or along) query bitmap	US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB	SAME	OFF	2006/08/23 12:23
S60	11	((("5560007") or ("6757670") or ("5666528") or ("5495608") or ("20040181514") or ("6996556") or ("6999958") or ("20040225639") or ("20050187917") or ("6778996") or ("5884307")).PN.	US-PGPUB; USPAT; USOCR	OR	OFF	2006/08/23 13:35
S62	3	query bitmap delay	US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB	SAME	OFF	2006/08/23 13:36
S61	1	query optimiz\$6 bitmap delay	US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB	SAME	OFF	2006/08/23 13:36
S4	8	Day-Paul-Reuben.IN. and (bitmap or (bit adj map))	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/08/23 17:04
S63	9	Day-Paul-Reuben.IN. and (bitmap or (bit adj map))	US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/08/23 17:05

EAST Search History

S64	5	Muras-Brian-robert.IN. and (bitmap or (bit adj map))	US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/08/23 17:07
S6	5	Muras-Brian-robert.IN. and (bitmap or (bit adj map))	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/08/23 17:07
S66	1	dynamic adj bitmap adj updating	US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/08/23 17:08
S65	70	(query adj optimization) and (bitmap or (bit adj map))	US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/08/23 17:08
S36	1	dynamic adj bitmap adj updating	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/08/23 17:08
S7	66	(query adj optimization) and (bitmap or (bit adj map))	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/08/23 17:08
S68	22	concurrent bitmap updating	US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB	SAME	OFF	2006/08/23 17:22

EAST Search History

S67	0	concurrent adj bitmap adj updating	US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/08/23 17:22
S70	21	query execution (generat\$ or creat\$) (Bitmap or (bit adj map)) and 707/1-4.ccls. and 707/100-101.ccls. and optimization	US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB	AND	OFF	2006/08/23 17:45
S69	0	query execution (generat\$ or creat\$) (Bitmap or (bit adj map)) and 707/1-4.ccls. and 707/100-101.ccls. and optimization	US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB	SAME	OFF	2006/08/23 17:45
S20	12	query and execution and (generat\$ or creat\$) same (Bitmap or (bit adj map)) and 707/1-4.ccls. and 707/100-101.ccls. and optimization	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/08/23 17:45
S72	0	IBM.AS. and (Bitmap or (bit adj map)) and 707/1-4.ccls. and 707/100-101.ccls.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/08/23 18:05
S71	0	IBM.AS. and (Bitmap or (bit adj map)) and 707/1-4.ccls. and 707/100-101.ccls. and optimization and (I/O or (input with output))	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/08/23 18:05
S23	0	IBM.AS. and (generat\$ or creat\$) same (Bitmap or (bit adj map)) and 707/1-4.ccls. and 707/100-101.ccls. and optimization and (I/O or (input with output))	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/08/23 18:05
S74	16	"international business machines". AS. and (Bitmap or (bit adj map)) and 707/1-4.ccls. and 707/100-101.ccls.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/08/23 18:06

EAST Search History

S73	819522	international business machines AS. and (Bitmap or (bit adj map)) and 707/1-4.ccls. and 707/100-101.ccls.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/08/23 18:06
S76	0	"international business machines". AS. and query adj execution same (Bitmap or (bit adj map)) and 707/1-4.ccls. and 707/100-101.ccls.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/08/23 18:08
S75	7	"international business machines". AS. and query same (Bitmap or (bit adj map)) and 707/1-4.ccls. and 707/100-101.ccls.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/08/23 18:08
S80	37061	"international business machines"". AS. and" "707".clas.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/08/23 18:09
S79	37061	"international business machines"". AS.and" "707".clas.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/08/23 18:09
S78	0	"international business machines". AS. and day-paul.in. and "707".clas.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/08/23 18:09
S77	0	"international business machines". AS. and day-paul.in. and 707/1-4. ccls. and 707/100-101.ccls.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/08/23 18:09
S82	206	"international business machines". AS. and "707".clas. and (bitmap or (bit adj map))	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/08/23 18:10

EAST Search History

S81	5220	"international business machines". AS. and "707".clas.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/08/23 18:10
S84	3	"international business machines". AS. and 707/5.ccls. and (bitmap or (bit adj map)) same (concurrent or while or parallel or along)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/08/23 18:15
S83	39	"international business machines". AS. and "707".clas. and (bitmap or (bit adj map)) same (concurrent or while or parallel or along)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/08/23 18:15
S37	0	query near5 plan and dynamic and bitmap same ((S36 or one or active) and ("0" or zero or inactive)) and statistic	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/08/23 18:24
S86	0	query same (concurrent\$ or while or along or parallel or inconjunction) same (bitmap or (bit adj map)) same ((S85 or one or active) and ("0" or zero or inactive)) and statistic	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/08/23 18:25
S85	1	dynamic adj bitmap adj updating	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/08/23 18:25
S87	40128	(bitmap or (bit adj map))	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/08/23 18:26
S89	1	(bitmap or (bit adj map)) same (query adj execution) and delay	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/08/23 18:27

EAST Search History

S88	9	(bitmap or (bit adj map)) same (query adj execution)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/08/23 18:27
S91	369	(bitmap or (bit adj map)) same (query)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/08/23 18:29
S90	9	(bitmap or (bit adj map)) same (query adj execution)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/08/23 18:29
S93	1	(bitmap or (bit adj map)) same (query) same (while during along parallel concurrent\$3 inconjunction) same delay	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/08/23 18:30
S92	66	(bitmap or (bit adj map)) same (query) same (while during along parallel concurrent\$3 inconjunction)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/08/23 18:30
S95	21	(bitmap or (bit adj map)) same (query) same (while during along parallel concurrent\$3 inconjunction) same (delay optimiz\$5 speed fast\$2 time)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/08/23 18:31
S94	11	(bitmap or (bit adj map)) same (query) same (while during along parallel concurrent\$3 inconjunction) same (delay optimiz\$5 speed fast\$2)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/08/23 18:31

EAST Search History

Ref #	Hits	Search Query	DBs	Default Operator	Plurals	Time Stamp
L1	22	(execut\$7 run\$4 process\$3) near2 query same (generat\$3 creat\$3 build\$3) near3 bitmap	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/08/24 08:49



STIC Search Report

EIC 2100

STIC Database Tracking Number: 199586

TO: Michael Pham
Location: RND 3D18
Art Unit: 2167
Thursday, August 24, 2006

Case Serial Number: 10/660167

From: Lance Sealey
Location: EIC 2100
RND-4B11
Phone: 571-272-8666

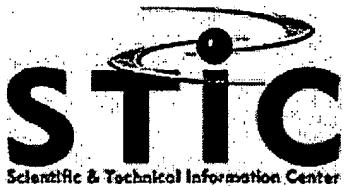
Lance.Sealey@uspto.gov

Search Notes

Dear Michael,

These were the closest results I could find. I found executing a query concurrently with building a bitmap, but not the underlined element in claim 1. Please let me know if you have any questions.

Lance



STIC EIC 2100

Search Request Form

199586

Today's Date:

8/23/2006

What date would you like to use to limit the search?

Priority Date: 9/11/2003 Other:

Name Michael Pham

AU 2167 Examiner # 81563

Room # 3D18 Phone 571-272-3924

Serial # 10660167

Format for Search Results (Circle One):

 PAPER DISK EMAIL

Where have you searched so far?

USP DWPI EPO JPO ACM IBM TDB

IEEE INSPEC SPI Other EAST / NPL

Is this a "Fast & Focused" Search Request? (Circle One) YES NO

A "Fast & Focused" Search is completed in 2-3 hours (maximum). The search must be on a very specific topic and meet certain criteria. The criteria are posted in EIC2100 and on the EIC2100 NPL Web Page at <http://ptoweb/patents/stic/stic-tc2100.htm>.

What is the topic, novelty, motivation, utility, or other specific details defining the desired focus of this search? Please include the concepts, synonyms, keywords, acronyms, definitions, strategies, and anything else that helps to describe the topic. Please attach a copy of the abstract, background, brief summary, pertinent claims and any citations of relevant art you have found.

Is this request for a BOARD of APPEALS case? (Circle One) YES NO

Topic - Bitmapping for faster query performance
INVENTION ACCESSES BITMAP FIRST
look for: executing a query concurrently with building a bitmap.

That is, before ~~when~~ a query maybe executed
before a bitmap can be completely built.

(SEARCH PREP TIME: 33 TERMINAL TIME: 135)

STIC Searcher LANCE, SEALEY Phone 2-8666

Date picked up 8/24/06 Date Completed 8/24/06

Set Items Description
 S1 1568299 SEARCH??? OR TARGET??? OR QUERY??? OR RETRIEV??? OR LOCAT-
 ??? OR MATCH??? OR FETCH??? OR REQUEST???

S2 8103696 CONCURR????? OR SYNC OR SYMBIOTIC OR PARALLEL OR SIMULTANE-
 OUS?? OR SYNCHRON????? OR SAME() TIME OR TANDEM OR TOGETHER OR
 COLLATERAL?? OR CONTEMPORANEOUS??

S3 44026 BITMAP????? OR BITSTRING? ? OR BIT() (MAP????? OR STRING? ?)

S4 177 S1(10N)S2(10N)S3

S5 17946519 PERFORM????? OR OPTIMIZ????? OR EFFICEN?? OR THROUGHPUT OR -
 EFFECTIV????? OR PRODUCTIV??? OR ACHIEV?????

S6 12188268 (DATA(W) (UNIT? ? OR BASE? ? OR BANK? ? OR MIN??? OR SET? ?
 OR WAREHOUS???) OR DATABASE? ? OR FILE? ? OR MEMORY OR MEMOR-
 IES OR DBMS? ? OR RDB? ? OR DATAMIN??? OR DATAWAREHOUS??? OR
 REPOSITORY OR TABLE? ? OR ARRAY? ? OR DATASET? ? OR VLDB? ? OR
 LDB? ? OR

S7 153 S4 AND S5 AND S6

S8 147 S7 AND (PY<2003 OR PD<20020911)

S9 91 RD (unique items)

S10 0 AU=((DAY, P? OR DAY P?) AND (MURAS B? OR MURAS, B?))

S11 55 (AU=(DAY, P? OR DAY P? OR MURAS B? OR MURAS, B?) AND S2) N-
 OT S9

S12 41 RD (unique items)

S13 30 S12 AND (PY<2003 OR PD<20020911)

? show files

File 275:Gale Group Computer DB(TM) 1983-2006/Aug 23
 (c) 2006 The Gale Group

File 47:Gale Group Magazine DB(TM) 1959-2006/Aug 23
 (c) 2006 The Gale group

File 16:Gale Group PROMT(R) 1990-2006/Aug 23
 (c) 2006 The Gale Group

File 624:McGraw-Hill Publications 1985-2006/Aug 24
 (c) 2006 McGraw-Hill Co. Inc

File 484:Periodical Abs Plustext 1986-2006/Aug W3
 (c) 2006 ProQuest

File 613:PR Newswire 1999-2006/Aug 24
 (c) 2006 PR Newswire Association Inc

File 813:PR Newswire 1987-1999/Apr 30
 (c) 1999 PR Newswire Association Inc

File 239:Mathsci 1940-2006/Oct
 (c) 2006 American Mathematical Society

File 370:Science 1996-1999/Jul W3
 (c) 1999 AAAS

File 696:DIALOG Telecom. Newsletters 1995-2006/Aug 23
 (c) 2006 Dialog

File 621:Gale Group New Prod.Annou.(R) 1985-2006/Aug 23
 (c) 2006 The Gale Group

File 674:Computer News Fulltext 1989-2006/Aug W1
 (c) 2006 IDG Communications

File 88:Gale Group Business A.R.T.S. 1976-2006/Aug 14
 (c) 2006 The Gale Group

File 369:New Scientist 1994-2006/Jul W4
 (c) 2006 Reed Business Information Ltd.

File 160:Gale Group PROMT(R) 1972-1989
 (c) 1999 The Gale Group

File 635:Business Dateline(R) 1985-2006/Aug 24
 (c) 2006 ProQuest Info&Learning

File 15:ABI/Inform(R) 1971-2006/Aug 24
 (c) 2006 ProQuest Info&Learning

File 9:Business & Industry(R) Jul/1994-2006/Aug 23
 (c) 2006 The Gale Group

File 13:BAMP 2006/Aug W2

(c) 2006 The Gale Group
File 810:Business Wire 1986-1999/Feb 28
(c) 1999 Business Wire
File 610:Business Wire 1999-2006/Aug 24
(c) 2006 Business Wire.
File 647:CMF Computer Fulltext 1988-2006/Oct W1
(c) 2006 CMF Media, LLC
File 98:General Sci Abs 1984-2005/Jan
(c) 2006 The HW Wilson Co.
File 148:Gale Group Trade & Industry DB 1976-2006/Aug 23
(c) 2006 The Gale Group
File 634:San Jose Mercury Jun 1985-2006/Aug 23
(c) 2006 San Jose Mercury News
File 636:Gale Group Newsletter DB(TM) 1987-2006/Aug 23
(c) 2006 The Gale Group

?

12675604 DATA
6646973 UNIT? ?
20401815 BASE? ?
7461324 BANK? ?
7570673 MIN???
9284197 SET? ?
945792 WAREHOUS???
603371 DATA(W) (((((UNIT? ? OR BASE? ?) OR BANK? ?) OR MIN???) OR
SET? ?) OR WAREHOUS???)
2610851 DATABASE? ?
5272471 FILE? ?
1442878 MEMORY
291734 MEMORIES
82101 DBMS? ?
9227 RDB? ?
2665 DATAMIN???
2928 DATAWAREHOUS???
169473 REPOSITORY
3148555 TABLE? ?
1174211 ARRAY? ?
28289 DATASET? ?
2026 VLDB? ?
1870 LDB? ?
21542 DATABANK? ?
313 OODB? ?
411151 DB? ?
S612188268 (DATA(W) (UNIT? ? OR BASE? ? OR BANK? ? OR MIN???) OR SET?
? OR WAREHOUS???) OR DATABASE? ? OR FILE? ? OR MEMORY OR
MEMORIES OR DBMS? ? OR RDB? ? OR DATAMIN???) OR
DATAWAREHOUS???) OR REPOSITORY OR TABLE? ? OR ARRAY? ? OR
DATASET? ? OR VLDB? ? OR LDB? ? OR DATABANK? ? OR OODB? ?
OR DB? ?

9/9/54 (Item 2 from file: 813)
DIALOG(R)File 813:PR Newswire
(c) 1999 PR Newswire Association Inc. All rts. reserv.

1186660 SFM067
**Oracle Optimizes Developer 2000(TM) for Oracle8(TM)-Based Multi-Platform,
Multi-Tier Enterprise Applications**

DATE: November 17, 1997 08:02 EST WORD COUNT: 896

REDWOOD SHORES, Calif., Nov. 17 /PRNewswire/ -- Oracle Corp. (Nasdaq: ORCL) today announced that Developer/2000(TM) 2.0, the latest release of Oracle's industry-leading client/server and Web development tool, extends support for the enterprise features of Oracle8(TM), Oracle's next generation database for network computing. The immediate availability of Developer/2000 Release 2.0 allows application developers to build powerful data warehousing and transaction processing applications leveraging the market-leading technology of Oracle8.

"The shipment of Developer/2000 2.0 reinforces Oracle's commitment to provide a complete, end-to-end technology foundation for network computing-from powerful, industry leading database servers to tightly integrated development tools and Web servers," said Mark Jarvis, vice president, System Products Marketing, Oracle. "This new release of Developer/2000 will let programmers leverage the breakthrough capabilities of Oracle8, delivering on Oracle's plan to distribute all of a company's data to all of its users faster and more affordably than ever before."

Oracle8 is a versatile information platform that enables high-speed transactions, better business decisions, and can substantially improve the productivity of organizations. Oracle8 is designed to lower an organization's computing costs, manage all types of data, and deliver faster information access to all kinds of users. Oracle8 provides several advances for the high-end OLTP and data warehousing markets such as support for mission-critical databases in the 100 terabyte range through data partitioning, 7X24 reliability and availability through transparent application failover and parallel server improvements, improved manageability through advanced server-managed backup and recovery, and scalability enhancements supporting tens of thousands of concurrent users.

Developer/2000 2.0 leverages the capabilities of the Oracle8 universal data server in a number of ways. The new release provides client-side support for several advances in SQL and PL/SQL, Oracle's powerful and easy to use procedural extension to industry standard SQL. With Developer/2000 2.0, application developers will be able to easily use several new PL/SQL features including PL/SQL Tables, Tables of Records, and Cursor variables. Developer 2000/2.0 simplifies PL/SQL application development with a number of easy-to-use wizards that automate the creation of an application through a point-and-click interface, drag-and-drop object partitioning, and transparent distributed debugging across client and server PL/SQL logic. Further, Developer/2000 Release 2.0 users can take advantage of the significant performance and scalability improvements that have been made in PL/SQL both with Oracle7(TM) and Oracle8.

With Developer/2000 2.0 and Oracle8, Oracle underscores its firm commitment to PL/SQL which offers enterprise application developers the fastest and easiest way to access SQL in the Oracle(R) database. "Oracle really listened to the requests of its users when building Developer/2000 Release 2.0," said Jeffrey M. Jacobs, president of Oracle's Development Tools User Group. "Our major enhancement requests-improved object

orientation, greater scalability, greater ease of learning and ease of use, enterprise reporting features, and distributed debugging-were all addressed in this release."

Developer/2000 2.0 can now take advantage of a broader range of Oracle8's transaction processing features, including the Oracle8 server's integrated advanced queuing capabilities. The move to more Web-based processing and distributed network computing requires applications that communicate with each other and guarantee continued processing without being directly connected. Oracle8 is the only database server that provides an asynchronous queuing facility which takes place directly within the database minimizing bottlenecks, improving response time and boosting application availability. Developer/2000 Release 2.0 allows OLTP application developers to take advantage of Oracle8's advanced queuing facility and a number of other performance and scalability improvements, including better memory and resource usage, connection multiplexing, and improved security through the Oracle Security Server's Cryptographic Toolkit.

Developer/2000 2.0 also provides unrivaled support for Oracle8 data warehousing features. Oracle8's patented parallel bitmap star query technologies (bitmap indexes, star queries, star joins) are now supported by Developer/2000, allowing multiple data warehousing functions to work in concert for dramatic performance gain. Additional parallelism capabilities can also be utilized including the data server's ability to conduct simultaneous insert, update and delete operations as well as the capability to run index scans, single partition scans and full table scans in parallel. Oracle8, which began shipping in June, continues to lead the market in database performance, scalability, reliability and ease-of-use on all platforms, including Windows NT. It is being used in production environments by a number of very large Oracle customers for a variety of mission-critical applications. Customer feedback indicates an easy migration path from Oracle7 to Oracle8, and rock-solid stability of the product.

Oracle Corporation is the world's leading supplier of software for information management, and the world's second largest software company. With annual revenues of \$6 billion, the company offers its database, tools and application products, along with related consulting, education and support services, in more than 140 countries around the world.

For more information about Oracle, call 650-506-7000. Oracle's World Wide Web address is (URL) <http://www.oracle.com/>.

Trademarks

Oracle is a registered trademark and Oracle7, Oracle8 and Developer/2000 are trademarks or registered trademarks of Oracle Corporation. All other products or company names mentioned are used for identification purposes only, and may be trademarks of their respective owners.

SOURCE Oracle Corp.

CONTACT: Paige O'Neill, 650-506-3429, or poneill@us.oracle.com, or Marlena Fernandez, 650-506-8565, or mcfernan@us.oracle.com, both of Oracle

Web site: <http://www.oracle.com>

(ORCL)

COMPANY NAME: ORACLE CORP.
TICKER SYMBOL: ORCL (NDQ)
PRODUCT: COMPUTER, ELECTRONICS (CPR); INTERNET, MULTIMEDIA,
 ONLINE (MLM)
DESCRIPTORS: NEW PRODUCTS & SERVICES (PDT)
STATE: CALIFORNIA (CA)
SECTION HEADING: BUSINESS; TECHNOLOGY

?

9/9/3 (Item 3 from file: 275)
DIALOG(R)File 275:Gale Group Computer DB(TM)
(c) 2006 The Gale Group. All rts. reserv.

02192438 SUPPLIER NUMBER: 20049363 (THIS IS THE FULL TEXT)
A DBA 's view of DB2 . (IBM's DB2 Universal Server 5.0 DBMS) (Server
Side) (Software Review) (Evaluation) (Column)
Rennhackkamp, Martin
DBMS, v10, n13, p71(4)
Dec, 1997
DOCUMENT TYPE: Evaluation Column ISSN: 1041-5173 LANGUAGE:
English RECORD TYPE: Fulltext; Abstract
WORD COUNT: 3494 LINE COUNT: 00282

ABSTRACT: Version 5.0 of IBM's DB2 Universal Server DBMS is a feature-rich package that is easy to install and administer. Database objects can be created and managed through DB2 's Control Center, which provides a GUI interface, or by using scripts via the Command Center or the Script Center. Version 5.0 is very tunable, particularly for multiple data locations and multiprocessor architectures. Version 5.0's most powerful capability is its extensible architecture, which adds Relational Extenders and component user-defined data types (UDDT) and user-defined functions (UDF), allowing users to extend the functionality of their DB2 servers to accommodate any application domain. DB2 Universal Server 5.0 raises the bar in extended-relational DBMS competition.

TEXT:

Taking IBM's new turbocharged DB2 version 5.0 out for a spin.

It was with great excitement that I opened the parcel, which arrived by courier from IBM. My first universal server had arrived -- in the form of IBM's DB2 Universal Server version 5 ("V5") for NT. The cover letter said "Check it out!" -- and that is what I plan to do in this month's column.

The package consisted of a Quick Beginnings Guide, various release notes, and a pile of CDs: DB2 Universal Server for NT, DB2 Universal Server Extenders, DB2 Universal Server Software Developers Kit, DB2 Universal Server Client Pack, Net.Data for NT, Visual Age for Basic, and Lotus Approach 97.

DB2 V5 combines what used to be Common Server (2.x) and the Parallel Edition (2.x). The Windows NT, Unix, and OS/2 code bases are the same, but other platforms, including the OS/390 and AS/400 versions of DB2 , have their own separate code bases.

Getting Started

Installing DB2 for NT is trivially simple. An Installation wizard asks you to select the target directory and the system components you want to install. It checks whether you have sufficient space (85MB for a typical installation) and whether your user name is acceptable (a valid user from the administrator group, not called Administrator). After it copies all the files , installs the services and program groups, and reboots your machine, you are set up to use DB2 Universal Server. After installation, a First Steps wizard guides you through setting up and using the sample database . You can also view the online documentation, called the "product library," from this wizard.

Tools and Facilities

DB2 V5 has a collection of new administration tools. The DB2 Control Center is an extensive GUI-based, Explorer-like tool with which you can manage everything from systems and databases to views, triggers, and users. You can perform most administrative tasks, such as configuring the system, managing directories, backing up and recovering databases , scheduling jobs, and managing storage media. The Control Center also contains a Replication Administration facility to set up graphically the

replication of data among systems. The Control Center, illustrated in Figure 1 (see page 72), consumes about 4MB of memory while running.

(Figure 1 ILLUSTRATION OMITTED)

The Command Center and Script Center provide interactive windows in which you can execute SQL statements, DB2 commands, and operating system commands. The results are displayed in a scrollable result window. From the Command Center, you can view the access plan and the statistics associated with a SQL statement before it executes. You can also schedule command scripts to run at a later date or at regular intervals. The results of scheduled jobs are written in a journal, which can be viewed from the script center.

The Performance Monitor provides information for performance analysis of the database manager and database applications.

Performance characteristics are captured periodically by the Snapshot Monitor. The snapshot data can be viewed in real-time as graphs or text and in detailed or summary form. You define exception conditions by specifying threshold values. When a threshold value is reached, any of the following actions can be initiated: notification through a window or an audible alarm, logging of a record in the database, or execution of a script or program. The Event Analyzer is used to capture summary information on the completion of events such as completed statements, completed transactions, and applications disconnected from the database.

The Governor is used to monitor applications running against the database. It consists of a front-end utility and a daemon. When you start the Governor's front-end utility, it starts the Governor daemon. The daemon collects statistics about the database applications. The Governor checks the statistics against rules specified in a configuration file. It can take various actions, such as changing the application's priority or even disconnecting it from the database.

Instances and Spaces

DB2, similar to Oracle, manages databases in the context of instances. An instance is a copy of the database manager code in a specific directory. You can have more than one instance on a system. DB2 builds a directory structure around the instance, where the database files are stored. During installation, DB2 creates a default instance called DB2. You can create additional instances, change the location of the instance's directory structure, and switch among instances by setting the appropriate environment variables. Instances are started and stopped from the DB2 Control Center. An instance must be running before you can connect to a database? precompile an application, or bind a package to the database. Prior to using an instance, you must, in DB2 terminology, "attach" to the instance using a valid username and password.

Unlike Oracle, where each database requires its own instance, a DB2 instance can manage many databases. But each database exists in the context of an instance. You can create a database in various ways. The easiest way is to use the Create Database SmartGuide wizard provided by the Control Center. The SmartGuide prompts you for the required details and manages the database creation. It has optional steps for space allocation and basic performance tuning. You can also create a database using the CREATE DATABASE command from the Command Center. This command provides all the options for aliases, code sets, collation sequences, extents, segment directories, and table spaces. Before gaining access to a database, you must connect using a valid username and password. Figure 2 illustrates the relationship between systems, instances, and databases.

(Figure 2 ILLUSTRATION OMITTED)

The objects associated with a database need not always be stored in the database directory. DB2 uses various storage space structures: containers, nodegroups, partitions, table spaces, and extents. A container is a name for a directory a device, or a file. A nodegroup is a

named set of one or more nodes belonging to a **database**. A nodegroup can contain one or more **database** partitions. You use **table** spaces to assign the locations of the **database** and its **tables** to containers. A **table** space is a storage structure containing **tables**, indexes, large objects, and **LONG** datatypes. **Table** spaces reside in nodegroups. The **table** space selected to hold a **table** defines how the **table** is spread across the **database** partitions in a nodegroup. A single **table** space can span several containers. The extent size for a **table** space indicates the number of pages of data that will be written to a container before switching to the next container.

A **database** must contain at least three **table** spaces. It must contain one catalog **table** space, which contains all the system catalog **tables** for the **database**. It must contain one or more user **table** spaces, which contain all the user-defined **tables**. Each **database** has a default **table** space, so if you do not specify a **table** space name when creating a **table**, it is placed in the first **table** space (or in the default **table** space if you haven't created any). Finally, a **database** must contain one or more temporary **table** spaces for temporary **tables**. If there is more than one temporary **table** space, temporary objects are allocated in a round-robin fashion.

If you are planning **database** installations of any substantial size, you need to know about containers, nodegroups, partitions, **table** spaces, extents, and their interrelationships. The DB2 Administrator's Guide provides useful information.

Database Objects

DB2 supports a complete set of **database** objects. Most **database** objects, including systems, instances, **databases**, **table** spaces, **tables**, views, aliases, triggers, schemes, indexes, connections, replication details, buffer pools, user-defined datatypes, user-defined functions, packages, users, and groups can be created and managed from the Control Center or the Command Center. Some objects, such as event monitors and procedures, must be created and managed from the Command Center.

Schemas are new in DB2 V5. It supports a complete implementation of schemes. A schema is a named logical grouping of **database** objects. When an object such as a **table**, view, alias, distinct type, function, index, package, or trigger is created, it is assigned to a schema. If you do not explicitly specify a schema, the default schema (identified by your user ID) is assumed. **Tables**, views, indexes, and grants on these can also be created as part of the CREATE SCHEMA command.

When you create a **table**, you have to assign it to a schema -- otherwise it is assigned to the default schema. Obviously, you also define the columns of the **table**, with their datatypes. DB2 supports INTEGER, SMALLINT, FLOAT, REAL, DOUBLE, DECIMAL, NUMERIC, CHARACTER (optionally FOR BIT DATA), VARCHAR (optionally FOR BIT DATA), LONG VARCHAR, BLOB, CLOB, DBCLOB, GRAPHIC, VARGRAPHIC, LONG VARGRAPHIC, DATE, TIME, TIMESTAMP, and user-defined distinct datatypes (UDDTS). A BLOB can be up to 2GB in size. You can have multiple BLOB columns, up to 4TB per **table**. CLOBS and DBCLOBS are special BLOBS for single-byte and double-byte character strings. There are various options you can specify for different columns, such as default values for nullable columns, compression for BLOB columns, and nonlogging for BLOB columns. You can specify different **table** spaces for the **table**, its BLOBS, and its indexes, optionally with partitioning -- otherwise everything is stored in the default **table** space.

DB2 also has an almost complete implementation of integrity constraints. A **table** can have named or unnamed primary key, unique, referential, and user-defined check constraints. A referential constraint can have NO ACTION, RESTRICT, CASCADE, or SET NULL options for deletes, but it can only have the RESTRICT options for updates.

DB2 supports a complete implementation of triggers. You can define

any number of triggers per **table**. The trigger specifies a set of actions that must be executed when an **INSERT**, **UPDATE**, or **DELETE** operation is performed on the base **table**. Triggers can fire before or after the operations on the base **table**. However, an operation executed before the operation on the base **table** will not cause other triggers to fire. A before-trigger may perform a **SELECT** statement, set variables, or signal a SQL state. An after-trigger may perform a **SELECT** statement, an **INSERT** statement, a qualified **UPDATE** statement (with a SQL WHERE clause), a qualified **DELETE** statement, or signal a SQL state. Triggers can fire per row or per statement. The old and new values of the row, for example, in an **UPDATE** statement, can be assigned different role names. A trigger can also include a **WHEN** clause to specify under which conditions it should fire. However, when you use triggers and constraints together, you should consider their execution sequences carefully. The interaction between triggers and constraints is explained very well in the **SQL Reference Manual**.

You can code **DB2** stored procedures in Java (using IBM's Visual Age for Java, available separately), Basic (through IBM's Visual Age for Basic), or in 3GLs such as Cobol, Fortran, C, and C++. A stored procedure can process multiple rows and return multiple result sets. The stored procedure is typically not stored in the **database**'s system catalogs but in a designated directory. This open and extensible server architecture makes your stored procedures independent of **DB2** releases.

UDDTs and UDFs

A user-defined distinct type (UDDT) is a new datatype derived from an existing datatype. It is defined using the **CREATE DISTINCT TYPE** statement, which names the new datatype, specifies its base datatype, and specifies whether comparisons between instances are allowed. Once you have created a UDDT you can use it to define columns in a **CREATE TABLE** statement.

DB2's UDDTs support strong typing, meaning that although a UDDT shares the same internal representation as the source type, the two types are incompatible. Instances of UDDTs cannot be used as arguments of functions or operations defined on the source type, not even for system-provided or built-in functions. Instances of the same UDDT can only be compared with each other if the **WITH COMPARISONS** clause was specified during creation. You must provide equivalent user-defined functions for the built-in functions. However, when you create a new UDDT based on an existing datatype, **DB2** generates the necessary type-casting functions between the UDDT and the existing datatype.

A user-defined function (UDF) is defined to extend the built-in SQL functions. It is created using the **CREATE FUNCTION** statement and can be called wherever a built-in function can. There are two types of UDFs: An external function is written in a programming language, while a sourced function's implementation is inherited from an existing function. External UDFs are classified as **table** or column functions. A column UDF can be used anywhere an expression is allowed, for example, in a **SELECT** statement, a **CREATE VIEW** statement, a **CHECK CONSTRAINT**, or in a trigger definition. A column function returns a single scalar value. A **table** UDF can only be used in a **FROM** clause -- it returns an entire **table** as a result. UDFs are stored in **DB2** in the same way as stored procedures.

I described UDDTs, user-defined datatypes (UDTs), and UDFs in the context of **DB2**'s Relational Extenders, comparing them to Informix's DataBlades and Oracle's Cartridges, in my feature article this month, "Extending Relational DBMSs" (page 45). Briefly, Relational Extenders are collections of UDDTs and UDFs, used to extend the functionality of the DBMS to cater to non-traditional data. **DB2** Universal Server V5 is released with four built-in Relational Extenders: text, image, video, and audio.

Replication

DB2 V5's replication facilities are the result of bundling in what

used to be a separate IBM product called Data Propagator. DB2's replication facilities copy modified data from one source database to one or more target databases. Replication is managed from the Control Center, which includes facilities for defining source and target tables, scheduling, data enhancement, and conflict detection. To designate a table for replication, it must use the DATA CAPTURE CHANGES option, which indicates that extra information for replication must be logged. This is specified as part of the CREATE TABLE or ALTER TABLE statements.

Replication is performed by two programs: Capture and Apply. The Capture program takes changes to the source tables, as recorded in the database log, and stores them until the target is ready to receive them. The Apply program replicates the changes to the target databases. The data is replicated asynchronously, either pushed from the source to one or more targets or pulled off the source from the target. The Apply server can transform the data it is replicating.

DB2 can manage update-anywhere replication, allowing updates to the source or target tables in a replication configuration. You can specify transaction conflicts through declarative constraints or triggers, and you can resolve conflicts through compensating transactions.

In V5, various performance improvements have also been made to the replication mechanisms, including the use of stored procedures and static SQL, reduced connect processing, and uncommitted reads during the recording process.

Concurrency Control

Like most other relational DBMSs, DB2 uses locking as its concurrency control method. DB2 can lock rows, tables, and table spaces, with automatic escalation from row- to table-level locks, based on the availability of lock resources.

DB2 V5 employs four isolation levels: repeatable read, read stability, cursor stability, and uncommitted read. The isolation level determines how data is locked from other processes while the data is being accessed. The isolation level is in effect for the duration of a transaction. DB2 uses various lock modes to implement these isolation levels. The default lock modes can be overridden per transaction with the LOCK TABLE statement.

OLAP Support

DB2 Universal Server V5 has various new features for OLAP, including star joins, cubes, roll-up, parallel querying, a cost-based optimizer, query rewriting, and bitmap indexes. DB2 uses bitmap indexes when processing warehouse-type queries. It builds the bitmap indexes dynamically during query processing and uses them during query execution. It applies a similar technique when processing star joins from large fact tables.

There are new extensions to the GROUP BY clause of the SQL SELECT statement for roll-up processing and to form cubes. You can add so-called super groups to the GROUP BY clause, which in turn can consist of ROLLUP and CUBE clauses, with additional grouping columns. The effect of a ROLLUP clause is similar to a control break in report writers. The effect of a CUBE is equivalent to a cross tabulation.

Performance

DB2 V5 incorporates a number of performance enhancement features; many of these aim to exploit multiprocessor architectures. It employs a technique called intraquery parallelism, where complex queries are decomposed into subqueries that are executed in parallel on multiple processors. This suits symmetrical multiprocessor (SMP) architectures, where a number of agents can process the query on shared disks and in shared memory. Massively parallel processor (MPP) and cluster architectures are exploited through hash-based table partitioning or through databases partitioned across clusters. Other improvements aimed at multiprocessor architectures include parallel I/O parallel index creation, parallel load, and parallel backup and restore utilities.

Other performance improvements to DB2 V5 include memory usage through multiple and tunable buffer pools that can be assigned to different table spaces, large memory support with 64-bit addressing, asynchronous writes, prefetch reading, dynamic space allocation, big block reads, and the Global SQL Cache.

Extensible and Extensive

After this evaluation of DB2 V5 for NT, my initial enthusiasm only waned for one reason: the vastness of its functionality, which can be tiring to evaluate. DB2 V5 is an extensive product. In this month's column I could only review a certain number of features important to a typical DBA. However, I didn't mention its rich security features, such as directory services, DCE security services, security roles, single login, and the use of Kerberos technology. Nor did I mention its backup and recovery facilities, such as table space-based and partition-based recovery, online backup and online restore, point-in-time recovery per table space, and faster restart.

DB2 V5 also includes many new application development features such as recursive queries for bill-of-material type structures; left, right, and full outer joins; UDFs; the Visual Explain utility to investigate the internal query access plans; and a call-level interface. This obviously also includes DB2 V5's extensive support for the Web, offering full support for Java. Applications developed in Java and Java applets can access a DB2 database directly through JDBC calls. You can also develop stored procedures, triggers, and functions in Java. Table -level UDFs, developed in Java, can be used to provide access to nonrelational data in the FROM clauses of SQL SELECT statements.

From a DBA's point of view, DB2 V5 is extremely easy to install and administer. Database objects can be created and managed either through the Control Center, with its convenient GUI interface, or by executing scripts through the Command Center or the Script Center. In an environment with CASE tools or where there are multiple databases, you may prefer to use scripts to ensure that the databases are consistent with one another. Scripts are also useful for development, test, and deployment life cycles. DB2 is highly tunable, especially for multiple data locations and multiprocessor architectures. The only additions I would like to see are permanent bitmap indexes and a choice of table and index storage structures. At this stage you can only store data in tables of 4K pages and indexes with B-trie structures.

I found DB2's most powerful feature by far to be its extensible architecture. Having added the Relational Extenders and the component UDDTs and UDFs, IBM is letting its users extend the functionality of their DB2 servers to cater to any application domain. The extended-relational DBMS competition just got tougher.

Martin Rennhackkamp is the owner and principal consultant of The Data Base Approach, a corporation specializing in relational and distributed databases, based in Cape Town, South Africa. You can email Martin at mr@dba.co.za or visit his Web site at www.dba.co.za.

COPYRIGHT 1997 M&T Publishing Inc.

COMPANY NAMES: International Business Machines Corp.--Products
DESCRIPTORS: DBMS ; Software Single Product Review
PRODUCT/INDUSTRY NAMES: 7372421 (DBMS)
SIC CODES: 7372 Prepackaged software
TICKER SYMBOLS: IBM
TRADE NAMES: DB2 Universal Server 5.0 (DBMS)--Evaluation
FILE SEGMENT: CD File 275

Set	Items	Description
S1	4366924	SEARCH??? OR TARGET??? OR QUERY??? OR RETRIEV??? OR LOCAT- ??? OR MATCH??? OR FETCH??? OR REQUEST???
S2	3732762	CONCURR????? OR SYNC OR SYMBIOTIC OR PARALLEL OR SIMULTANE- OUS?? OR SYNCHRON????? OR SAME()TIME OR TANDEM OR TOGETHER OR COLLATERAL?? OR CONTEMPORANEOUS??
S3	7403	BITMAP????? OR BITSTRING? ? OR BIT() (MAP????? OR STRING? ?)
S4	282	S1 AND S2 AND S3
S5	11736663	PERFORM????? OR OPTIMIZ????? OR EFFICEN?? OR THROUGHPUT OR - EFFECTIV????? OR PRODUCTIV??? OR ACHIEV?????
S6	3734643	(DATA(W) (UNIT? ? OR BASE? ? OR BANK? ? OR MIN??? OR SET? ? OR WAREHOUS???) OR DATABASE? ? OR FILE? ? OR MEMORY OR MEMOR- IES OR DBMS? ? OR RDB? ? OR DATAMIN??? OR DATAWAREHOUS??? OR REPOSITORY OR TABLE? ? OR ARRAY? ? OR DATASET? ? OR VLDB? ? OR LDB? ? OR
S7	177	S4 AND S5
S8	104	S7 AND S6
S9	58	RD (unique items)
S10	46	S9 AND (PY<2003 OR PD<20020911)
? show files		
File	2:INSPEC 1898-2006/Aug W2	(c) 2006 Institution of Electrical Engineers
File	6:NTIS 1964-2006/Aug W2	(c) 2006 NTIS, Intl Cpyrgh All Rights Res
File	8:Ei Compendex(R) 1970-2006/Aug W2	(c) 2006 Elsevier Eng. Info. Inc.
File	34:SciSearch(R) Cited Ref Sci 1990-2006/Aug W3	(c) 2006 The Thomson Corp
File	35:Dissertation Abs Online 1861-2006/Jun	(c) 2006 ProQuest Info&Learning
File	56:Computer and Information Systems Abstracts 1966-2006/Aug	(c) 2006 CSA.
File	57:Electronics & Communications Abstracts 1966-2006/Aug	(c) 2006 CSA.
File	60:ANTE: Abstracts in New Tech & Engineer 1966-2006/Aug	(c) 2006 CSA.
File	65:Inside Conferences 1993-2006/Aug 24	(c) 2006 BLDSC all rts. reserv.
File	94:JICST-EPlus 1985-2006/May W2	(c) 2006 Japan Science and Tech Corp (JST)
File	95:TEME-Technology & Management 1989-2006/Aug W3	(c) 2006 FIZ TECHNIK
File	99:Wilson Appl. Sci & Tech Abs 1983-2006/Jul	(c) 2006 The HW Wilson Co.
File	111:TGG Natl. Newspaper Index(SM) 1979-2006/Aug 11	(c) 2006 The Gale Group
File	144:Pascal 1973-2006/Jul W5	(c) 2006 INIST/CNRS
File	256:TecInfoSource 82-2006/Nov	(c) 2006 Info.Sources Inc
File	434:SciSearch(R) Cited Ref Sci 1974-1989/Dec	(c) 2006 The Thomson Corp
?		

7797541 DATA
1698401 UNIT? ?
8654642 BASE? ?
482278 BANK? ?
2483485 MIN???
2463054 SET? ?
56649 WAREHOUS???
460408 DATA(W) (((UNIT? ? OR BASE? ?) OR BANK? ?) OR MIN???) OR
SET? ?) OR WAREHOUS???)
683672 DATABASE? ?
290128 FILE? ?
698955 MEMORY
82345 MEMORIES
22666 DBMS? ?
1298 RDB? ?
426 DATAMIN???
170 DATAWAREHOUS???
36704 REPOSITORY
620704 TABLE? ?
803640 ARRAY? ?
58682 DATASET? ?
1448 VLDB? ?
811 LDB? ?
13048 DATABANK? ?
1428 OODB? ?
353726 DB? ?
S6 3734643 (DATA(W) (UNIT? ? OR BASE? ? OR BANK? ? OR MIN???) OR SET?
? OR WAREHOUS???) OR DATABASE? ? OR FILE? ? OR MEMORY OR
MEMORIES OR DBMS? ? OR RDB? ? OR DATAMIN???) OR
DATAWAREHOUS???) OR REPOSITORY OR TABLE? ? OR ARRAY? ? OR
DATASET? ? OR VLDB? ? OR LDB? ? OR DATABANK? ? OR OODB? ?
OR DB? ?

10/9/33 (Item 1 from file: 34)
DIALOG(R) File 34:SciSearch(R) Cited Ref Sci
(c) 2006 The Thomson Corp. All rts. reserv.

11125576 Genuine Article#: 610FP Number of References: 28
Title: Parallel Star Join plus DataIndexes: Efficient query processing
in data warehouses and OLAP
Author(s): Datta A (REPRINT) ; VanderMeer D; Ramamritham K
Corporate Source: Georgia Inst Technol,Atlanta//GA/30332 (REPRINT); Georgia
Inst Technol,Atlanta//GA/30332; Indian Inst Technol,Bombay
400076/Maharashtra/India/
Journal: IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, 2002 , V14,
N6 (NOV-DEC), P1299-1316
ISSN: 1041-4347 Publication date: 20021100
Publisher: IEEE COMPUTER SOC, 10662 LOS VAQUEROS CIRCLE, PO BOX 3014, LOS
ALAMITOS, CA 90720-1314 USA
Language: English Document Type: ARTICLE
Geographic Location: USA; India
Journal Subject Category: COMPUTER SCIENCE, ARTIFICIAL INTELLIGENCE;
COMPUTER SCIENCE, INFORMATION SYSTEMS; ENGINEERING, ELECTRICAL &
ELECTRONIC
Abstract: On-Line Analytical Processing (OLAP) refers to the technologies
that allow users to efficiently retrieve data from the data
warehouse for decision-support purposes. Data warehouses tend to
be extremely large-it is quite possible for a data warehouse to be
hundreds of gigabytes to terabytes in size [3]. Queries tend to be
complex and ad hoc, often requiring computationally expensive
operations such as joins and aggregation. Given this, we are interested
in developing strategies for improving query processing in data
warehouses by exploring the applicability of parallel processing
techniques. In particular, we exploit the natural partitionability of a
star schema and render it even more efficient by applying DataIndexes-a
storage structure that serves both as an index as well as data and
lends itself naturally to vertical partitioning of the data.
Dataindexes are derived from the various special purpose access
mechanisms currently supported in commercial OLAP products.
Specifically, we propose a declustering strategy which incorporates
both task and data partitioning and present the Parallel Star Join
(PSJ) Algorithm, which provides a means to perform a star join in
parallel using efficient operations involving only rowsets and
projection columns. We compare the performance of the PSJ Algorithm
with two parallel query processing strategies. The first is a
parallel join strategy utilizing the Bitmap Join Index (BJI),
arguably the state-of-the-art OLAP join structure in use today. For the
second strategy we choose a well-known parallel join algorithm,
namely the pipelined hash algorithm. To assist in the performance
comparison, we first develop a cost model of the disk access and
transmission costs for all three approaches. Performance comparisons
show that the Dataindex-based approach leads to dramatically lower disk
access costs than the BJI, as well as the hybrid hash approaches, in
both speedup and scaleup experiments, while the hash-based approach
outperforms the BJI in disk access costs. With regard to transmission
overhead, our performance results show that PSJ and BJI outperform
the hash-based approach. Overall, our parallel star join algorithm
and dataindexes form a winning combination.
Descriptors--Author Keywords: parallel star join ; OLAP ; query
processing ; dataindexes

Cited References:

- *RED BRICK SYST, 1995, STAR SCHEM STARJ TEC
- *T PROC PERFCOUNC, 1997, TPC BENCHM D DEC SUP

ABDELGUERFI M, 1998, PARALLEL DATABASE TE
BALLINGER C, 1997, P29, P 8 INT HONG KONG CO
CHAURET N, 1997, V26, P1, DRUG METAB DISPOS
CHEN IR, 1995, V7, P4, IEEE T KNOWL DATA EN
CHEN MS, 1996, V8, IEEE T KNOWLEDGE DAT
DATTA A, 1998, TR9807 U AR
DATTA A, 2001, TR2001 15 APPL PARAL
DEWITT D, 1992, V35, P6, COMMUN ACM
DEWITT DJ, 1992, P27, P 18 INT C VER LARG
ENGELBERT S, 1989, 894 TAND COMP
FERNANDEZ PM, 1994, P492, P SIGMOD 94
FRENCH CD, 1997, P194, PROC INT CONF DATA
HALLMARK G, 1997, P314, PROC INT CONF DATA
HARINARAYAN V, 1996, P205, P ACM SIGMOD INT C M
INMON WH, 1996, BUILDING DATA WAREHO
KITSUREGAWA M, 1995, P101, P JSPS NUS SEM COMP
LEE C, 1993, P411, P 9 INT C DAT ENG VI
LO ML, 1993, P ACM SIGMOD
LU H, 1994, QUERY PROCESSING PAR
LYONS K, 1997, HOSTING MASSIVE DATA
ONEIL P, 1997, P38, P ACM SIGMOD C MAN D
ONEIL P, 1995, V24, P8, SIGMOD REC
SCHNEIDER D, 1989, P ACM SIGMOD JUN
SEETHA M, 1990, V2, P410, IEEE T KNOWL DATA EN
SHATDAL A, 1993, P119, P 1993 ACM SIGMOD C
WOLF JL, 1991, P78, P 1 INT C PAR DISTR

?

10/9/18 (Item 18 from file: 2)
DIALOG(R) File 2:INSPEC
(c) 2006 Institution of Electrical Engineers. All rts. reserv.

06242611 INSPEC Abstract Number: C9605-6160D-016
Title: The Microsoft relational engine
Author(s): Graefe, G.
Author Affiliation: Microsoft Corp., Redmond, WA, USA
Conference Title: Proceedings of the Twelfth International Conference on Data Engineering (Cat. No.96CB35888) p.160-1
Editor(s): Su, S.Y.W.
Publisher: IEEE Comput. Soc. Press, Los Alamitos, CA, USA
Publication Date: 1996 **Country of Publication:** USA xx+678 pp.
ISBN: 0 8186 7240 4 **Material Identity Number:** XX96-00892
U.S. Copyright Clearance Center Code: 1063-6382/96/\$5.00
Conference Title: Proceedings of the Twelfth International Conference on Data Engineering
Conference Sponsor: IEEE Comput. Soc. Tech. Committee on Data Eng
Conference Date: 26 Feb.-1 March 1996 **Conference Location:** New Orleans, LA, USA
Language: English **Document Type:** Conference Paper (PA)
Treatment: General, Review (G)
Abstract: Microsoft offers three very successful database products, FoxPro, Access and SQL Server. While SQL Server excels in multi-user transaction performance , Access and its underlying Jet engine excel as an end-user and development tool for desktop and client-server applications. One of our top priorities is to improve the integration of these two products. With respect to their query processing capabilities, we plan on combining the strengths of SQL Server with those of Access. SQL Server's strengths are focused on management of very large tables , server-side cursors, and the use of stored procedures as scripts and as triggers. Access strengths are queries over multiple servers, updatable query results, and bit - mapped processing. In our next generation of products, SQL Server will employ new query processing technology. Both optimization and execution will be based on an extensible set of operators. We are focusing on the relational algebra augmented with a few operators such as the top operator found in Access, and suitable loops-, index-, sort-, hash-, and bitmap -based execution algorithms. Moreover, we are planning on executing these algorithms both sequentially and in parallel . We have four design goals, namely functionality, performance , scalability, and extensibility. (0 Refs)
Subfile: C
Descriptors: parallel algorithms; query languages; query processing ; relational algebra; relational databases ; SQL
Identifiers: Microsoft relational engine; FoxPro; Access; SQL Server; multi-user transaction performance ; Jet engine; end-user tool; development tool; client-server applications; desktop applications; query processing; very large table management; server-side cursors; stored procedures; updatable query results; bit - mapped processing; execution ; optimization ; relational algebra; top operator; execution algorithms; sequential algorithms
Class Codes: C6160D (Relational databases); C4240P (Parallel programming and algorithm theory)
Copyright 1996, IEE

10/9/16 (Item 16 from file: 2)

DIALOG(R)File 2:INSPEC

(c) 2006 Institution of Electrical Engineers. All rts. reserv.

06484864 INSPEC Abstract Number: C9703-6160Z-003

Title: Data warehousing features in Informix OnLine XPS

Author(s): Sundaresan, P.

Conference Title: Proceedings of the Fourth International Conference on Parallel and Distributed Information Systems (Cat. No.96TB100085) p.288

Publisher: IEEE Comput. Soc. Press, Los Alamitos, CA, USA

Publication Date: 1996 Country of Publication: USA xi+295 pp.

ISBN: 0 8186 7475 X Material Identity Number: XX96-03187

U.S. Copyright Clearance Center Code: 0 8186 7475 X/96/\$5.00

Conference Title: Proceedings of 4th International Conference on Parallel and Distributed Information Systems

Conference Sponsor: IEEE Compt. Soc. Tech. Committee on Data Eng.; ACM SIGMOD

Conference Date: 18-20 Dec. 1996 Conference Location: Miami Beach, FL, USA

Language: English Document Type: Conference Paper (PA)

Treatment: Practical (P)

Abstract: Summary form only given, as follows. The data warehousing application domain is an important area of focus for Informix's OnLine XPS massively parallel server. Fast query processing is a central requirement in this domain. Use of indexes has traditionally been an important query processing technique, helping to reduce response times and increase throughput. The data warehousing environment, characterized by its load-query-refresh mode of operation, offers even greater scope for use of indexes. The article describes three new indexing related features in OnLine XPS which together provide significant performance benefits in a wide variety of situations. Bitmap indexes, along with multi-index scans, provide orders-of-magnitude improvement for queries typified by the set query benchmark. Pushdown semi-joins combine the benefits of multi-index scans with the scalability of hash joins to efficiently process star-joins. Finally, generalized-key indexes expand the notion of what can be an index key and provide the ability to store various pre-computed results in an index. Optimizer extensions allow these features to be used in a mix-n-match fashion, thus maximizing the benefits of these features while minimizing the need for user level directives. (0 Refs)

Subfile: C

Descriptors: file servers; indexing; parallel processing; query processing; very large databases

Identifiers: Informix OnLine XPS massively parallel server; data warehousing ; fast query processing; load-query-refresh mode; indexes; performance benefits; bitmap indexes; multi-index scans; set query benchmark; pushdown semi-joins; hash joins; star-joins; generalized-key indexes; pre-computed results; optimizer extensions; user level directives

Class Codes: C6160Z (Other DBMS)

Copyright 1997, IEE

10/9/15 (Item 15 from file: 2)

DIALOG(R)File 2:INSPEC

(c) 2006 Institution of Electrical Engineers. All rts. reserv.

06566857 INSPEC Abstract Number: C9706-7102-010
Title: Teaching an OLTP database kernel advanced datawarehousing techniques

Author(s): French, C.D.

Author Affiliation: Sybase Inc., Burlington, MA, USA

Conference Title: Proceedings. 13th International Conference on Data Engineering (Cat. No.97CB36038) p.194-8

Publisher: IEEE Comput. Soc. Press, Los Alamitos, CA, USA

Publication Date: 1997 Country of Publication: USA xvii+592 pp.

ISBN: 0 8186 7807 0 Material Identity Number: XX97-00832

U.S. Copyright Clearance Center Code: 1063-6382/97/\$10.00

Conference Title: Proceedings 13th International Conference on Data Engineering

Conference Sponsor: IEEE Comput. Soc

Conference Date: 7-11 April 1997 Conference Location: Birmingham, UK

Language: English Document Type: Conference Paper (PA)

Treatment: Practical (P)

Abstract: Most, if not all, of the major commercial database products available today were written more than 10 years ago. Their internal designs have always been heavily optimized for OLTP applications. Over the last couple of years as DSS and data warehousing have become more important, database companies have attempted to increase their performance with DSS-type applications. Most of their attempts have been in the form of added features like parallel table scans and simple bitmap indexing techniques. These were chosen because they could be quickly implemented (1-2 years), giving some level of increased query performance . The paper contends that the real performance gains for the DSS application have not yet been realized. The performance gains for DSS will not come from parallel table scans, but from major changes to the low level database storage management used by OLTP systems. One Sybase product, Sybase-IQ has pioneered some of these new techniques. The paper discusses a few of these techniques and how they could be integrated into an existing OLTP database kernel. (0 Refs)

Subfile: C

Descriptors: decision support systems; query processing; relational databases ; transaction processing; very large databases

Identifiers: OLTP database kernel teaching; advanced data warehousing techniques; commercial database products; decision support systems; parallel table scans; bitmap indexing techniques; query performance ; low level database storage management; Sybase-IQ; on-line transaction processing

Class Codes: C7102 (Decision support systems); C6160Z (Other DBMS); C6130 (Data handling techniques); C6160D (Relational databases)

Copyright 1997, IEE

10/9/2 (Item 2 from file: 2)

DIALOG(R)File 2:INSPEC

(c) 2006 Institution of Electrical Engineers. All rts. reserv.

08591216 INSPEC Abstract Number: C2003-05-6160B-008

Title: Database management system support for architecture with intelligent memories

Author(s): Gejdos, I.; Ben Miled, Z.; Liu, J.; Bukhres, O.

Author Affiliation: Electr. Eng. Dept., Indiana Univ., Indianapolis, IN, USA

Conference Title: Proceedings of the ISCA 15th International Conference Parallel and Distributed Computing Systems p.199-205

Editor(s): Smari, W.W.; Guizani, M.

Publisher: Int. Soc. Comput. & their Applications, Cary, NC, USA

Publication Date: 2002 Country of Publication: USA vii+562 pp.

Material Identity Number: XX-2002-02296

Conference Title: PDCS 2002: 15th International Conference on Parallel and Distributed Computing Systems

Conference Sponsor: Int. Soc. Comput. & their Applications

Conference Date: 19-21 Sept. 2002 Conference Location: Louisville, KY, USA

Language: English Document Type: Conference Paper (PA)

Treatment: Practical (P)

Abstract: By adding logic to the memory , intelligent memories allow the creation of a cost- effective parallel architecture that can address the computational needs of data intensive applications. Currently, database applications are the most predominant data intensive applications because database instances continue to increase at a rapid rate. This paper introduces database management techniques designed to take advantage of intelligent memory in order to meet the performance needs of current massive database applications. In particular bitmap indexing, which is very common in current database management systems, is optimized for intelligent memories . Results suggest that the use of intelligent memories can improve query processing in large databases .

(15 Refs)

Subfile: C

Descriptors: database indexing; parallel architectures; parallel databases ; parallel memories ; query processing; very large databases

Identifiers: intelligent memories ; cost- effective parallel architecture; data intensive applications; database management system support; performance needs; massive database applications; bitmap indexing; query processing; large databases

Class Codes: C6160B (Distributed databases); C5220P (Parallel architecture); C6160Z (Other DBMS); C6120 (File organisation)

Copyright 2003, IEE

Set	Items	Description
S1	2189314	SEARCH??? OR TARGET??? OR QUERY??? OR RETRIEV??? OR LOCAT- ??? OR MATCH??? OR FETCH??? OR REQUEST???
S2	3042677	CONCURR????? OR SYNC OR SYMBIOTIC OR PARALLEL OR SIMULTANE- OUS?? OR SYNCHRON????? OR SAME()TIME OR TANDEM OR TOGETHER OR COLLATERAL?? OR CONTEMPORANEOUS??
S3	12738	BITMAP????? OR BITSTRING? ? OR BIT() (MAP????? OR STRING? ?)
S4	560	S1 AND S2 AND S3
S5	16	AU=((DAY, P? OR DAY P?) AND (MURAS B? OR MURAS, B?))
S6	0	S5 NOT AD=(20020911:20060824)
S7	29	(AU=(DAY, P? OR DAY P? OR MURAS B? OR MURAS, B?) AND S2) N- OT S5
S8	26	S7 NOT AD=(20020911:20060824)
S9	9	(S4 AND (MC=T01 OR IC=(G06F-007/00))) NOT (S5 OR S8)
S10	78	(S1(10N)S2(10N)S3) NOT (S5 OR S8:S9)
S11	69	S10 NOT AD=(20020911:20060824)
S12	36	S11 AND (S1/TI OR S2/TI OR S3/TI)
S13	3868860	PERFORM????? OR OPTIMIZ????? OR EFFICEN?? OR THROUGHPUT OR - EFFECTIV????? OR PRODUCTIV??? OR ACHIEV?????
S14	231	(S4 AND S13) NOT (S5 OR S8:S9 OR S12)
S15	84	(S1(50N)S2(50N)S3(50N)S13) NOT (S5 OR S8:S9 OR S12)
S16	45	(S1(25N)S2(25N)S3(25N)S13) NOT (S5 OR S8:S9 OR S12)
S17	37	S16 NOT AD=(20020911:20060824)

? show files

File 347:JAPIO Dec 1976-2005/Dec (Updated 060404)
(c) 2006 JPO & JAPIO

File 350:Derwent WPIX 1963-2006/UD=200654
(c) 2006 The Thomson Corporation

5/5/14 (Item 14 from file: 350)
DIALOG(R) File 350:Derwent WPIX
(c) 2006 The Thomson Corporation. All rts. reserv.

0014893339 - Drawing available
WPI ACC NO: 2005-241082/
XRPX Acc No: N2005-198706

Background process e.g. input/output activity, index bitmapping method for database management system, involves producing bitmap by initially setting units to active value, and selectively setting respective units to inactive value

Patent Assignee: INT BUSINESS MACHINES CORP (IBMC)

Inventor: DAY P R ; MURAS B R

Patent Family (1 patents, 1 countries)

Patent Application

Number	Kind	Date	Number	Kind	Date	Update
US 20050060293	A1	20050317	US 2003660167	A	20030911	200525 B

Priority Applications (no., kind, date): US 2003660167 A 20030911

Patent Details

Number	Kind	Lan	Pg	Dwg	Filing Notes
US 20050060293	A1	EN	10	3	

Alerting Abstract US A1

NOVELTY - The method involves executing a database query (46) on table entries in a table using a bitmap having a respective unit associated with each table entry. The bitmap is generated by initially setting units of the bitmap to an active value and selectively setting respective units to an inactive value. The respective units are associated with entries that do not satisfy a portion of the query.

DESCRIPTION - INDEPENDENT CLAIMS are also included for the following:

1.a computer-readable medium having instructions, when executed by processors, to cause the processors to perform a method of reducing input/output (I/O) activity while executing a database query

2.an apparatus for executing a query.

USE - Used for bitmapping a background process e.g. I/O activity, index in a database management system that is utilized for accessing information stored in a database for various applications e.g. commercial, industrial, technical, scientific and educational applications.

ADVANTAGE - Utilizes the bitmap to reduce unnecessary I/O operation and match records back to a user without the typical delay period needed to build the complete bitmap. Allows the user to receive immediate feedback for submitting the query and enjoys the performance enhancement of using the bitmap.

DESCRIPTION OF DRAWINGS - The drawing shows a block diagram illustrating the components and flow of information between the components in a database management system.

32Database management system

40SQL parser

42Optimizer

44Database engine

46Database query

Title Terms/Index Terms/Additional Words: BACKGROUND; PROCESS; INPUT; OUTPUT; ACTIVE; INDEX; METHOD; DATABASE; MANAGEMENT; SYSTEM; PRODUCE; INITIAL; SET; UNIT; VALUE; SELECT; RESPECTIVE; INACTIVE

Class Codes

International Classification (Main) : G06F-007/00

US Classification, Issued: 707003000

File Segment: EPI;

DWPI Class: T01

Manual Codes (EPI/S-X) : T01-F05A; T01-J05B3; T01-S01B; T01-S03

Set	Items	Description
S1	2593939	SEARCH??? OR TARGET??? OR QUERY??? OR RETRIEV??? OR LOCAT- ??? OR MATCH??? OR FETCH??? OR REQUEST???
S2	1509235	CONCURR????? OR SYNC OR SYMBIOTIC OR PARALLEL OR SIMULTANE- OUS?? OR SYNCHRON????? OR SAME()TIME OR TANDEM OR TOGETHER OR COLLATERAL?? OR CONTEMPORANEOUS??
S3	16312	BITMAP????? OR BITSTRING? ? OR BIT() (MAP????? OR STRING? ?)
S4	175	S1(10N)S2(10N)S3
S5	3	S4 AND IC=(G06F-007/00)
S6	1499162	PERFORM????? OR OPTIMIZ????? OR EFFICEN?? OR THROUGHPUT OR - EFFECTIV????? OR PRODUCTIV??? OR ACHIEV?????
S7	1531097	(DATA(W) (UNIT? ? OR BASE? ? OR BANK? ? OR MIN?? OR SET? ? OR WAREHOUS???) OR DATABASE? ? OR FILE? ? OR MEMORY OR MEMOR- IES OR DBMS? ? OR RDB? ? OR DATAMIN?? OR DATAWAREHOUS?? OR REPOSITORY OR TABLE? ? OR ARRAY? ? OR DATASET? ? OR VLDB? ? OR LDB? ? OR
S8	55	(S4(100N)S6(100N)S7) NOT S5
S9	33	S8 NOT PD=(20020911:20060824)

? show files

File 348:EUROPEAN PATENTS 1978-2006/ 200634
(c) 2006 European Patent Office

File 349:PCT FULLTEXT 1979-2006/UB=20060817,UT=20060810
(c) 2006 WIPO/Univentio

?

733024 DATA
905458 UNIT? ?
1360655 BASE? ?
64462 BANK? ?
767500 MIN???
1013163 SET? ?
10065 WAREHOUS???
119715 DATA(W) (((((UNIT? ? OR BASE? ?) OR BANK? ?) OR MIN???) OR
SET? ?) OR WAREHOUS???)
122657 DATABASE? ?
980172 FILE? ?
328536 MEMORY
47023 MEMORIES
1521 DBMS? ?
691 RDB? ?
129 DATAMIN???
28 DATAWAREHOUS???
11906 REPOSITORY
625316 TABLE? ?
230046 ARRAY? ?
4131 DATASET? ?
112 VLDB? ?
1195 LDB? ?
1941 DATABANK? ?
150 OODB? ?
97736 DB? ?
S7 1531097 (DATA(W) (UNIT? ? OR BASE? ? OR BANK? ? OR MIN???) OR SET?
? OR WAREHOUS???) OR DATABASE? ? OR FILE? ? OR MEMORY OR
MEMORIES OR DBMS? ? OR RDB? ? OR DATAMIN???) OR
DATAWAREHOUS???) OR REPOSITORY OR TABLE? ? OR ARRAY? ? OR
DATASET? ? OR VLDB? ? OR LDB? ? OR DATABANK? ? OR OODEB? ?
OR DB? ?

YOU CAN DECIDE IF IT IS WORTH PRINTING OUT THE REFERENCE
TO WHICH THIS REFERS (355 PAGES).

9/3, K/29 (Item 5 from file: 349)

DIALOG(R) File 349:PCT FULLTEXT

(c) 2006 WIPO/Univentio. All rts. reserv.

00293338 **Image available**

DATABASE USING TABLE ROTATION AND BIMAPPED QUERIES

BASE DE DONNEES A ROTATION DE TABLES ET A INTERROGATIONS EN MODE POINT

Patent Applicant/Assignee:

FDC INC,

Inventor(s):

EMERSON Michael Gene,

WESTMAN Kelly Reed,

PILLAI Sushil,

Patent and Priority Information (Country, Number, Date):

Patent: WO 9511487 A1 19950427

Application: WO 94US12074 19941024 (PCT/WO US9412074)

Priority Application: US 93141285 19931022

Designated States:

(Protection type is "patent" unless otherwise stated - for applications prior to 2004)

CA GB

Publication Language: English

Fulltext Word Count: 85937

Fulltext Availability:

Detailed Description

Detailed Description

... DecMessageQue (DMQ) is a messaging system that reliably moves information from one computer to another. **Database Link**" uses this product to get information from the PC 10 to the server 14...

...data

that is needed is "mapped" to a global secz:icn and the operating system **optimizes** the access to this information. Because this operating system is very efficient at this type of memory **optimization** and swapping, this approach is very fast. It also allows an inherent multi-threading (more than one type of supporting process happening at the **same time**) that enables disk I/O to happen at the **same time** as data is evaluated at the beginning of a process.

Data Base Link evaluates data...Link" server, essentially, each record is removed, rotated by 90 degrees, and placed within the **memory** of the **Database** Link TM server 14. Rather than having all data stored contiguously for each individual customer...

...shown in

FIGURE 1, the data for each field across all records/customers in the **database** is stored contiguously as illustrated in FIGURE 3. For example, instead of having columns that...that is meaningful to the requester of information. In these types of cases, several **bitmaps** must be combined **together** to form some type of aggregated result. Figure 11 illustrates one method of combining multiple **bitmaps** resulting from a

complex **query** .

For example, for the **query** select "all males who live in the state of New York", two bitmaps can be...

...data is stored in chunks of 64 bits. This allows for one CPU instruction to **perform** the equivalent of 64 AND operations if they were @)erformed on a record by record 64 bit processing **performed** by the computer speeds up the process of determining whether a Darticular customer satisfies the query while using less **memory** . In contrast, standard relational **database** processing examines an entire record to determine if the query is satisfied, and then proceeds...

...this real world situation, by storing a "ring" of the 20 (or whatever number optimizes **performance** with the resources available) most recent queries that have been used by a particular user To further **optimize** the **performance** of the overall system, more sophisticated weighting algorithms similar to those used in contemporary operating...

...a single CPU instruction can operate on 5 the bits corresponding to 64 customers in **parallel** . if b4tmaps 41-48 are stored on the ring, and a new **query** L **matches** **bitmap** 43, then this new **query** is not recalculated. The elimination of this processing further reduces search time.

Relational Structure
Database...

...rectangular file, not several files joined together in a more real world representation of data. **Database Link**' actually stores inverted indices to gain access across multiple data **tables** .

Complex Bitmar) Processin
Customer Level

As shown in FIGURE 4 **Database Link**" involves a number of different **file** structure types. In the embodiment illustrated in FIGURE 4 there are a number of different...Item Queries

Line item queries are much more limited in scope than purchase queries.

The **Query Type**. The only standard **query** type is any.line **query** . This **query** groups all line items **together** into one batch and **performs** a simple query. This would be identical to making any-activity query with a specific...

?